Introduction to Smart Pointers Solutions

Copyright © James Raynard 2021

Traditional Pointers

- Briefly describe the traditional C++ pointer
  - The traditional C++ pointer is a memory address
- Briefly explain why using traditional C++ pointers is undesirable
  - Too easy to make mistakes which cause program crashes, data corruption etc
  - Manual management of allocated memory complicates programs and is error-prone

Copyright © James Raynard 2021

Smart Pointers

- What is a "smart pointer"?
  - A smart pointer manages its own allocated memory
- Describe briefly how a smart pointer manages memory
  - Memory is allocated by the smart pointer's constructor and released in its destructor

Copyright © James Raynard 2021

Smart Pointers

- Why are smart pointers considered preferable to traditional pointers?
  - Avoid the need to write memory management code
  - The pointer is always valid
  - Encapsulation prevents the pointer from being overwritten
  - No memory leaks
  - Transfer of ownership is correctly handled - avoids need for "deep copying"

Copyright © James Raynard 2021